TP ozone : synthèse

Résumé

Itération sur plusieurs échantillons et comparaison des erreurs de prévision.

1 Objectif

L'objectif final est d'arriver à une comparaison fine et synthétique des différentes méthodes pour aboutir à des prises de décision :

- Quelle méthode utiliser pour prévoir au mieux la concentration d'ozone ?
- Quelle stratégie et quelle méthode utiliser pour prévoir le dépassement du seuil :
  - Prévision quantitative puis compariaosn au seuil ?
  - Prévision qualitative ?

La taille de l'échantillon test, autour de 200 jours, est relativement modeste pour espérer une bonne précision sur la foi d'un seul échantillon. Pour préciser la comparaison c'est-à-dire pour prendre en compte la variance de l'estimation de l'erreur, le processus est itérée.

2 Itérations de l'estimation des erreurs

Consulter le programme `comparaison_ozone.R`, définir les paramètres : initialisation du générateur (xxx) et nombre d'itérations (xx au moins 30) avant de le faire exécuter en "batch" ou de nuit.

Ce programme lit les données "ozone.dat" et itère N fois le tirage d'un échantillon test pour estimer N erreurs de prévision pour chacune des méthodes de modélisation considérée. Sont calculées sur chaque échantillon test :

- l'erreur quadratique de prévision en régression,
- le taux d'erreur de classification issue de la prévision de dépassement de seuil,
- le taux d'erreur de classification comme prévision d'une variable binaire de dépassement.

Ces résultats sont respectivement stockés dans les matrices `res.reg`, `res.clas.r` et `res.clas.q`.

Pour construire les cournes ROC, le programme stocke également les prévisions pour chacune des méthodes de modélisation considérée et pour chaque échantillon test dans des variables de type liste et de nom : `list.methode` où méthode est la méthode de modélisation considérée. On se limite aux méthodes apparues les "meilleures" lors de la comparaison des distributions des erreurs. Attention, cette sélection n'est pas forcément "optimale" ; elle pourrait être complétée utilement.

3 Synthèses des résultats

3.1 Prévison de concentration

Calculer, comparer les moyennes et écarts-types des distributions des erreurs quadratiques de prévision. Tracer les diagrammes boîtes parallèles de ces distributions :

```
mean(res.reg)
boxplot(res.reg)
```

Commentaires.

3.2 Prévision de dépassement

Calculer, comparer les moyennes et écarts-types des distributions des taux d'erreur de prévision du dépassement de seuil. Tracer les diagrammes boîtes parallèles de ces distributions :

```
apply(res.clas.r,2,mean)
apply(res.clas.q,2,mean)
boxplot(data.frame(res.clas.r,res.clas.q))
```

Commentaires.

Tracer les courbes ROC par échantillon test en superposant des diagrammes boîtes visualisant les dispersions des courbes :

```
library(ROCR)
#création des objets ROC
pred <- prediction(list.svmr$predictions,
```

```
   list.svmr$labels)
perf.svmr <- performance(pred,"tpr","fpr")
plot(perf.svmr,col="grey82",lty=3)
plot(perf.svmr,lwd=3,avg="vertical",
   spread.estimate="boxplot",add=TRUE)

pred <- prediction(list.svmq$predictions,
   list.svmq$labels)
perf.svmq <- performance(pred,"tpr","fpr")
plot(perf.svmq,col="grey82",lty=3)
plot(perf.svmq,lwd=3,avg="vertical",
   spread.estimate="boxplot",add=TRUE)

pred <- prediction(list.rfr$predictions,
   list.rfr$labels)
perf.rfr <- performance(pred,"tpr","fpr")
plot(perf.rfr,col="grey82",lty=3)
plot(perf.rfr,lwd=3,avg="vertical",
   spread.estimate="boxplot",add=TRUE)

pred <- prediction(list.rfq$predictions,
   list.rfr$labels)
perf.rfq <- performance(pred,"tpr","fpr")
plot(perf.rfq,lty=3)
plot(perf.rfq,lwd=3,avg="vertical",
   spread.estimate="boxplot",add=TRUE)

pred <- prediction(list.log$predictions,
   list.log$labels)
perf.log <- performance(pred,"tpr","fpr")
plot(perf.log,lty=3)
plot(perf.log,lwd=3,avg="vertical",add=TRUE)

pred <- prediction(list.mlq$predictions,
   list.log$labels)
perf.mlq <- performance(pred,"tpr","fpr")
plot(perf.mlq,col="grey82",lty=3)
plot(perf.mlq,lwd=3,avg="vertical",
   spread.estimate="boxplot",add=TRUE)
```

Superposer les moyennes par méthode de ces courbes ROC.

```
plot(perf.mlq,col=1,avg="vertical")
plot(perf.log,col=2,avg="vertical",add=TRUE)
plot(perf.rfq,col=3,avg="vertical",add=TRUE)
plot(perf.rfr,col=4,avg="vertical",add=TRUE)
plot(perf.svmr,col=5,avg="vertical",add=TRUE)
plot(perf.svmq,col=6,avg="vertical",add=TRUE)
legend("bottomright",legend=c("acova","logit",
   "rfq","rfr","svmr","svmq"),col=1:6,pch="_")
```

Adapter ces graphiques afin de mettre clairement en évidence la “meilleure” méthode. Préciser ce choix en fonction d’un taux de faux positifs jugé acceptable : 10%, 20%, 40% en comparaison du taux de vrais positifs espéré.